R codes for: “Diversity loss from multiple interacting disturbances is regime-dependent.’’

L. Lear, H. Inamine, K. Shea, and A. Buckling

Updated: September 01, 2023

1 Preamble: Define functions, load data, and prepare data.

```r
# Preamble

rm(list=ls())

library(tidyverse)
library(vegan)
library(lme4)
library(gridExtra)
library(ggfortify)
library(car)

theme_set(theme_minimal())

res.bac = c("p", "o", "s", "a", "v")
all.bac = c(res.bac, "inv", "total.res", "total.all")


# Insert a ggplot layer under existing ones.
`-.gg` <- function(plot, layer) {
    if (missing(layer)) {
        stop("Cannot use `-.gg()` with a single argument. Did you accidentally put - on a new line?")
    }
    if (!is.ggplot(plot)) {
        stop('Need a plot on the left side')
    }
    plot$layers = c(layer, plot$layers)
    plot
}

# To format the scientific notation in the plots.
scientific_10 <- function(x) {
  ifelse(x==0, "0",
    parse(text=gsub("e\\+*", " %*% 10^", scales::scientific_format()(x)))
  )
}

# Function to calculate derived metrics from the raw data.
prepare.data = function(Data, DetectionLimit=0){
  # Replace NA with 0 for calculating statistics and plotting data.
  Data[is.na(Data)] = 0
  # Replace "0"s with lower detection limit.
  Data[Data==0] = DetectionLimit
  # Transform time-bewteen-disturbance event to disturbance frequency.
```

```
  Data$disturbance = 1/Data$disturbance
  # Put in dilutions used in the experiments.
  Data = cbind(Data, plate_dilution = 10^-5,
    microcosm_volume_ml = 6,
    glycerol_dilution = 0.5,
    plated_volume_ul = 25) %>%
    mutate(total_dilution = plate_dilution * glycerol_dilution *
      plated_volume_ul / (microcosm_volume_ml * 1000))

  Data.prop.all = Data[,c(res.bac, "inv")]/rowSums(Data[,c(res.bac, "inv")])
  Data.prop.res = Data[,c(res.bac)]/rowSums(Data[,c(res.bac)])

  names(Data.prop.all) = paste(c(res.bac, "inv"), ".prop.all", sep="")
  names(Data.prop.res) = paste(c(res.bac), ".prop.res", sep="")

  # Combine together the different measures.
  Data = cbind(Data, Data.prop.all, Data.prop.res,
    simpson = diversity(Data[,res.bac], index="simpson"),
    shannon = diversity(Data[,res.bac], index="shannon"),
    alpha = rowSums(Data[, res.bac] > DetectionLimit),
    simpson.inv = diversity(Data[, c(res.bac, "inv")], index="simpson"),
    shannon.inv = diversity(Data[, c(res.bac, "inv")], index="shannon"),
    alpha.inv = rowSums(Data[, c(res.bac, "inv")] > DetectionLimit))
  Data = cbind(Data,
    # Hill number of order 1
    D1 = exp(Data[, "shannon"]),
    D1.inv = exp(Data[, "shannon.inv"]))
  Data = cbind(Data,
    # Scaled Hill number of order 1
    D1.scaled = (Data[, "D1"] - 1)/(length(res.bac)-1),
    D1.inv.scaled = (Data[, "D1.inv"] - 1)/(length(res.bac)+1-1))
  return(Data)
}


autoplot.custom = function(fit){
  autoplot(fit, alpha=0.5,
    smooth.colour="red",
    label.colour="red")
}

# ggplot settings.
pub.layer = list(
  theme(panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    axis.title.y = element_text(size=12),
    axis.title.x = element_text(size=12),
    strip.background = element_blank(),
    strip.text = element_text(angle = 0, hjust = 0.0, size = 12),
    axis.text.x = element_text(size = 12),
    axis.text.y = element_text(size = 9),
    legend.position= 'right'),
    labs(colour ='Invaded',
    fill ='Invaded',
    linetype ='Invaded',
    alpha ='Invaded',
    shape ='Invaded'))

# ggplot settings.
```

```
fig.layer = list(
  scale_color_manual(values=c("no"="black", "yes"="red")),
  scale_fill_manual(values=c("no"="black", "yes"="red")),
  scale_x_continuous(breaks=1/c(2,4,8,16), labels=c("2\nHigh", "4", "8", "16\nLow")),
  xlab("Pulse frequency (every x days)"),
  theme_bw(),
  guides(color=guide_legend(override.aes = list(alpha=1))))


# New data used to plot model predictions.
newdat = expand_grid(
  disturbance = seq(1/16, 1/2, length.out=100),
  invaded = c("no", "yes"),
  total_dilution = 1)

# Load the data.
data.diversity = read.csv("data.csv", comment.char="#") %>%
  prepare.data(Data=.)
```

2 The effects of `invasion` and `pulse frequency` on the effective number of species of the resident community.

Here we perform regressions on the scaled effective numbers of species of the resident community samples. Resident communities in our experiment can contain 1 to 5 microbial species. We measure and scale the effective numbers of species of the samples so the scaled numbers range from 0 (only 1 species) to 1 (equal abundance of all 5 species), using the following formula:

$$\hat{D} = \frac{D-1}{D_{max}-1} = \frac{D-1}{4} \tag{1}$$

where D is the effective number of species and D_{max} is the maximum possible number of species, i.e. 5.

We run logistic regressions with quasibinomial family on $\hat{D}$ to assess the effects of `invasion` and `pulse frequency`:

$$logit(\hat{D}) \sim (\text{pulse} + \text{pulse}^2) \times \text{invasion}, \tag{2}$$

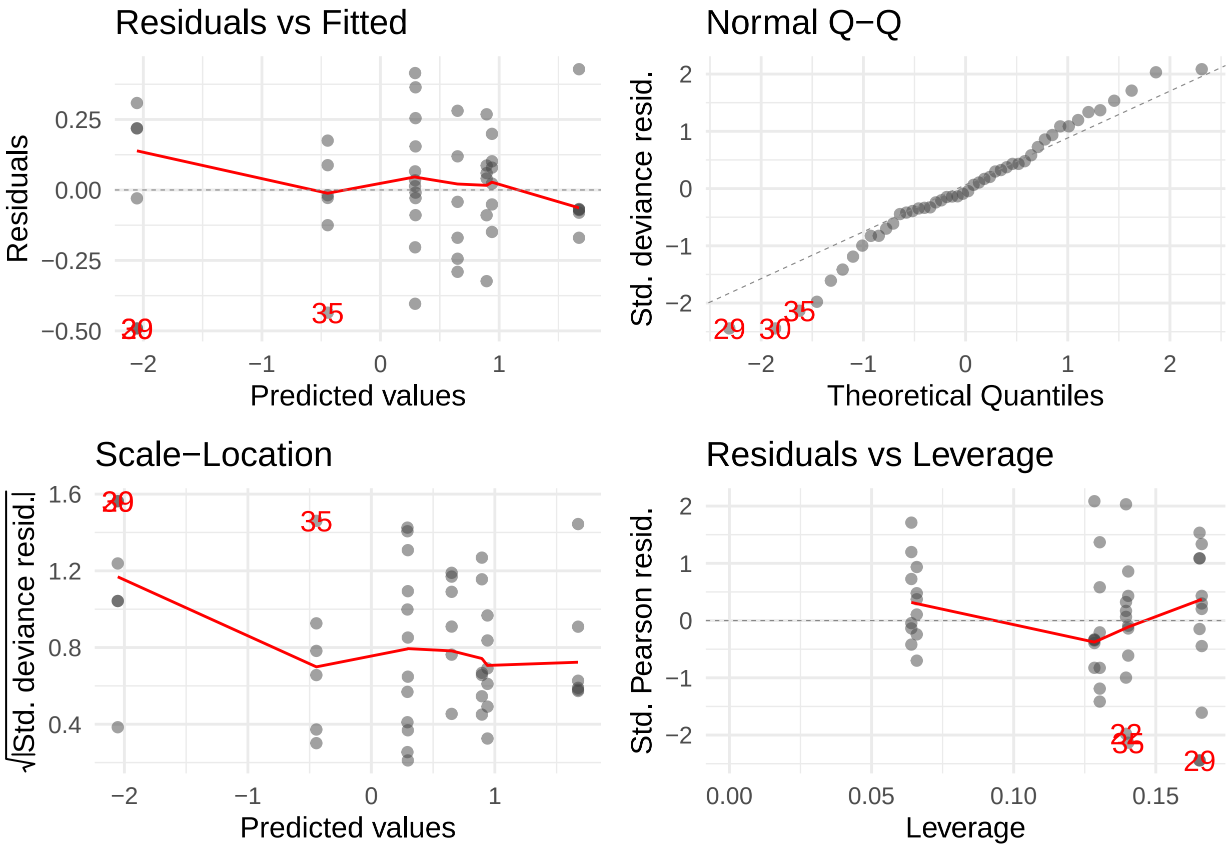
where "pulse" is the pulse frequency (*e.g.*, 1/16 for the regime with a pulse every 16 d) and "invasion" is a categorical variable coded as 0 for non-invaded and 1 for invaded samples.

```
fig.D1.scaled = ggplot(data=data.diversity,
  mapping=aes(x=disturbance, y=D1.scaled, color=invaded, shape=invaded, fill=invaded)) +
  geom_jitter(alpha=0.3, size=3.5, position=position_jitter(width=0.02, seed=1)) +
  geom_text(aes(label=alpha), position=position_jitter(width=0.02, seed=1), size=3) +
  scale_y_continuous(breaks=(1:5 - 1)/(4), labels=1:5) +
  fig.layer +
  pub.layer +
  ylab("Effective number of species")

# Logistic regression.
fit.D1.scaled = glm(D1.scaled ~ (disturbance + I(disturbance^2))*invaded,
  data=data.diversity, family="quasibinomial")

drop1(fit.D1.scaled, test="LRT")
```

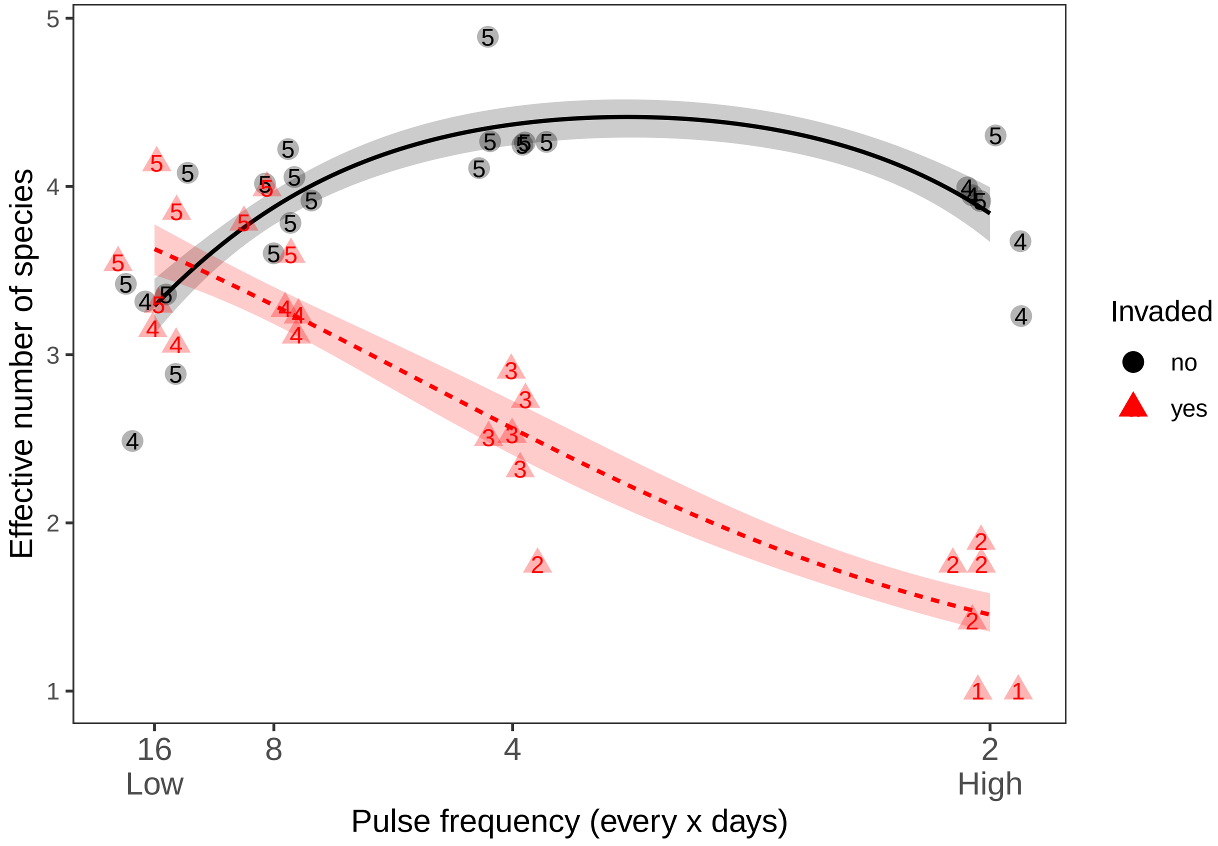
```
## Single term deletions
##
## Model:
## D1.scaled ~ (disturbance + I(disturbance^2)) * invaded
##                       Df Deviance scaled dev.  Pr(>Chi)
## <none>                     2.3112
## disturbance:invaded    1   3.3716     21.8927 2.883e-06 ***
```

```
## I(disturbance^2):invaded  1   2.7352      8.7552  0.003087 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Quasibinomial, so no AIC output.
# The full model has the smallest deviance.

# Model diagnostics.
autoplot.custom(fit.D1.scaled)
```



```
# The Q-Q plot shows some deviance, but overall not too bad...


# Make a dataframe to calculate the model predictions +/- the standard errors.
pred = predict(fit.D1.scaled, newdata=newdat, type="link", se.fit=T) %>%
  magrittr::extract(c("fit", "se.fit")) %>%
  as.data.frame() %>%
  cbind(newdat, .) %>%
  transform(p.fit=plogis(fit), p.lwr=plogis(fit-se.fit), p.upr=plogis(fit+se.fit))

# FIGURE 2
# Plot the model predictions onto the data points.
fig.D1.scaled +
  geom_line(data=pred, aes(y=p.fit, linetype=invaded), show.legend=F, size=0.75) +
  geom_ribbon(data=pred, aes(y=NULL, ymin=p.lwr, ymax=p.upr), show.legend=F, color=NA, alpha=0.2)
```



```r
ggsave("D1scaled.png")
```

```
## Saving 6.5 x 4.5 in image
```

```r
# The model output.
summary(fit.D1.scaled)
```

```
##
## Call:
## glm(formula = D1.scaled ~ (disturbance + I(disturbance^2)) *
##     invaded, family = "quasibinomial", data = data.diversity)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -0.49125  -0.09844  -0.01401   0.12817   0.42841
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   -0.5432     0.3215  -1.690 0.098528 .
## disturbance                   14.8621     3.3489   4.438 6.45e-05 ***
## I(disturbance^2)             -23.9719     5.7303  -4.183 0.000143 ***
## invadedyes                     1.5370     0.4433   3.467 0.001226 **
## disturbance:invadedyes       -20.2806     4.4416  -4.566 4.30e-05 ***
## I(disturbance^2):invadedyes   22.6185     7.7323   2.925 0.005531 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for quasibinomial family taken to be 0.0484361)
##
##     Null deviance: 11.8885  on 47  degrees of freedom
## Residual deviance:  2.3112  on 42  degrees of freedom
## AIC: NA
##
```

```
## Number of Fisher Scoring iterations: 5
```

```
# Calculate McFadden's pseudo R-squared:
##  1 - (deviance)/null.deviance
with(fit.D1.scaled, 1 - (deviance)/null.deviance)
```

```
## [1] 0.8055964
```

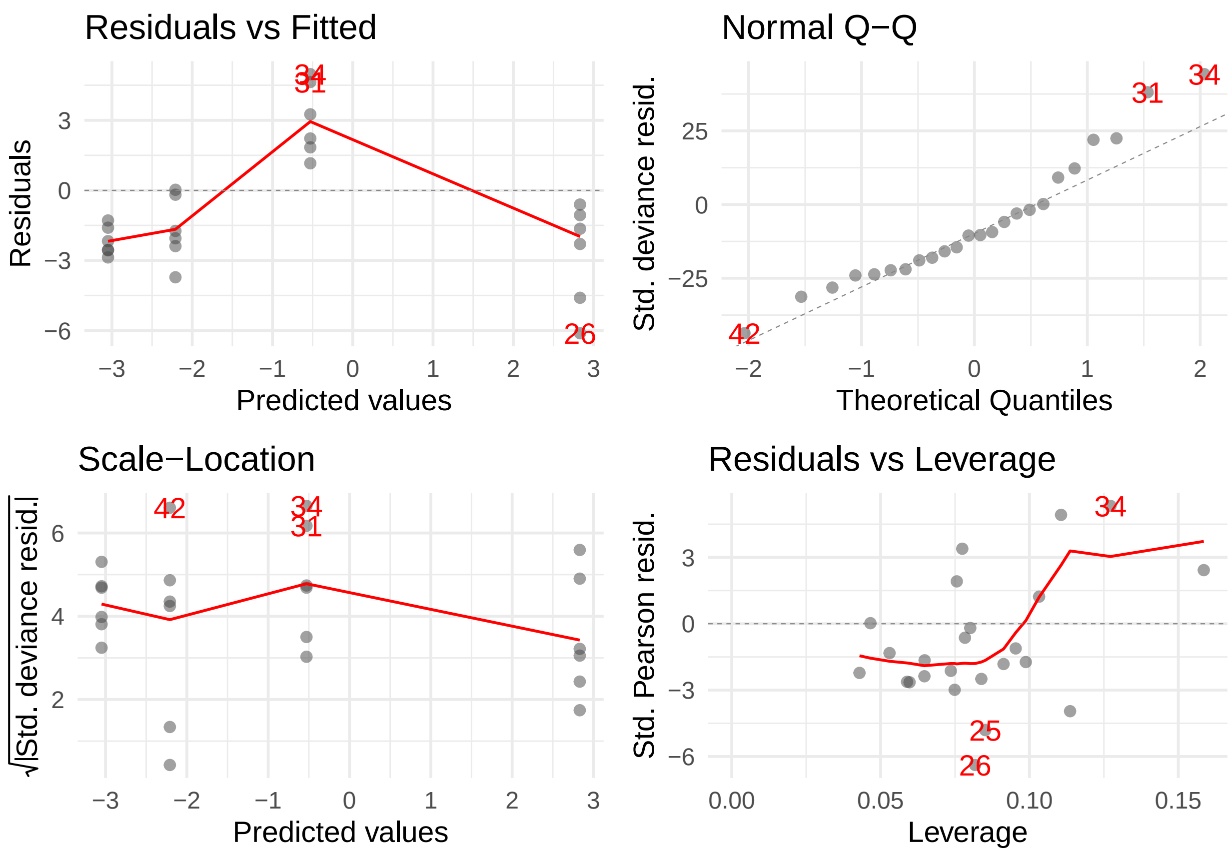
It looks like `invasion`, `pulse frequency`, and `pulse frequency`×`invasion` affect the resident diversity.

3 The effects of `pulse frequency` on invader fitness.

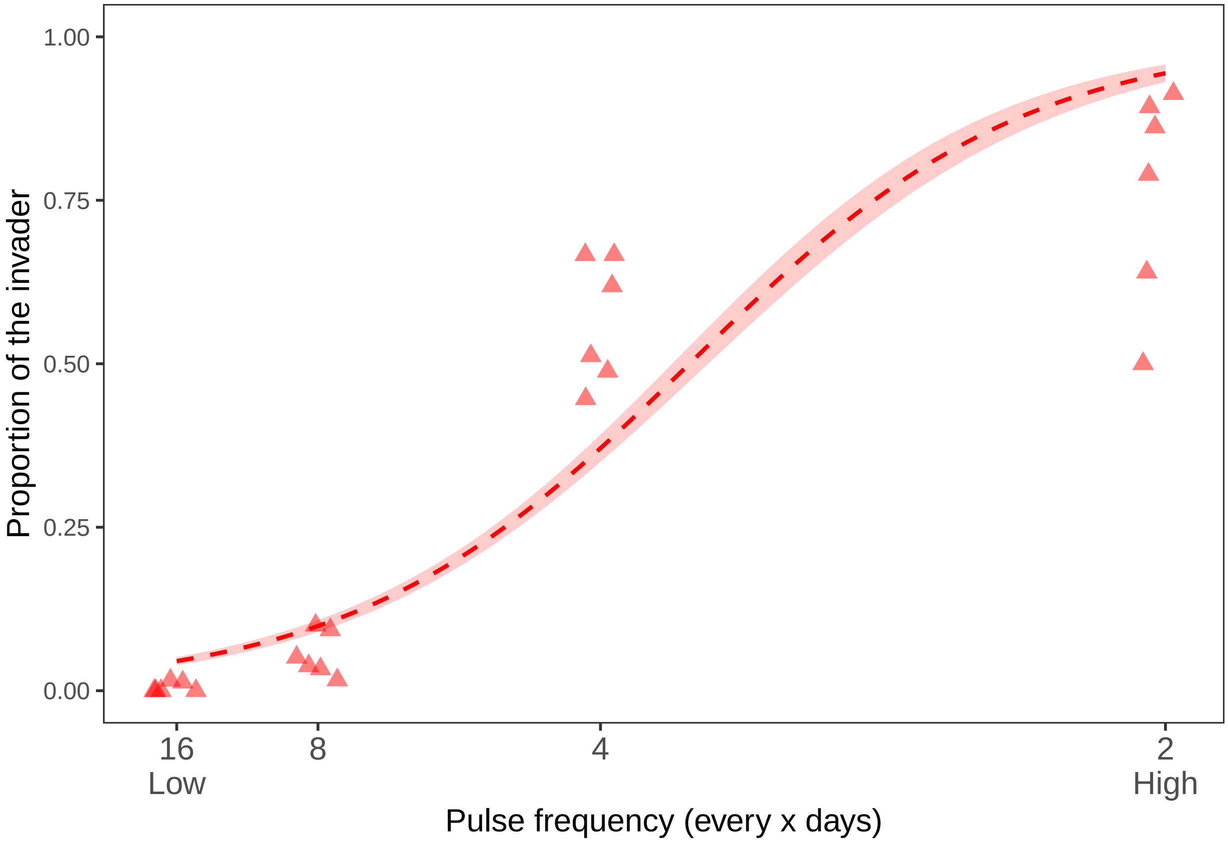
3.1 Proportion of the invader.

Here we assess the effects of `pulse frequency` on the proportion of the invader in the community.

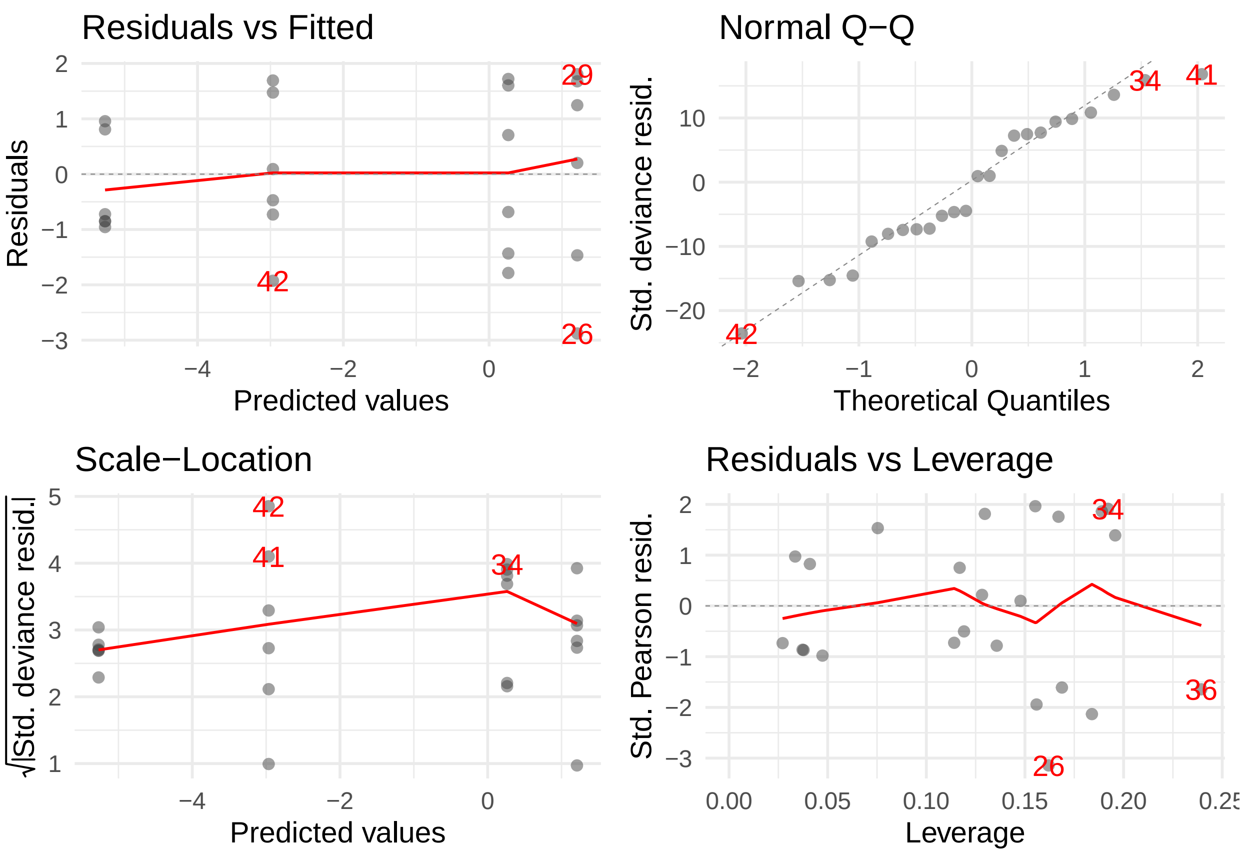
```
fig.inv.prop = ggplot(data=subset(data.diversity, invaded=="yes"),
 mapping=aes(x=disturbance, y=inv.prop.all)) +
 # geom_boxplot(aes(group=disturbance), outlier.shape = NA, lwd = 0.5,
 #   width = 0.03, show.legend = F, color="red") +
 geom_jitter(width=0.01, height=0, alpha=0.5, size=2.5, color="red", shape=17, fill="red") +
 ylab("Proportion of the invader") +
 ylim(c(0,1))+
 fig.layer +
 pub.layer


# First attempt at the logistic regression.
fit.inv.prop = glm(
  cbind(inv, (p+o+s+a+v)) ~ disturbance,
  data = data.diversity,
  subset = (invaded=="yes"),
  family = "binomial")

# Model diagnostics.
autoplot.custom(fit.inv.prop)
```



```
# The model fit doesn't look great. The residual plot shows possible quadratic effect.

# Plot the model.
# Make a dataframe to calculate the model predictions +/- the standard errors.
pred.inv.prop = predict(fit.inv.prop, newdata=newdat, type="response", se.fit=T) %>%
  magrittr::extract(c("fit", "se.fit")) %>%
  as.data.frame() %>%
  cbind(newdat, .) %>%
  transform(upr=fit+se.fit, lwr=fit-se.fit)

fig.inv.prop +
  geom_line(data=pred.inv.prop, aes(y=fit),
    size=0.75, linetype=2, color="red") +
  geom_ribbon(data=pred.inv.prop, aes(y=NULL, ymin=lwr, ymax=upr),
    alpha=0.2, color=NA, fill="red")
```



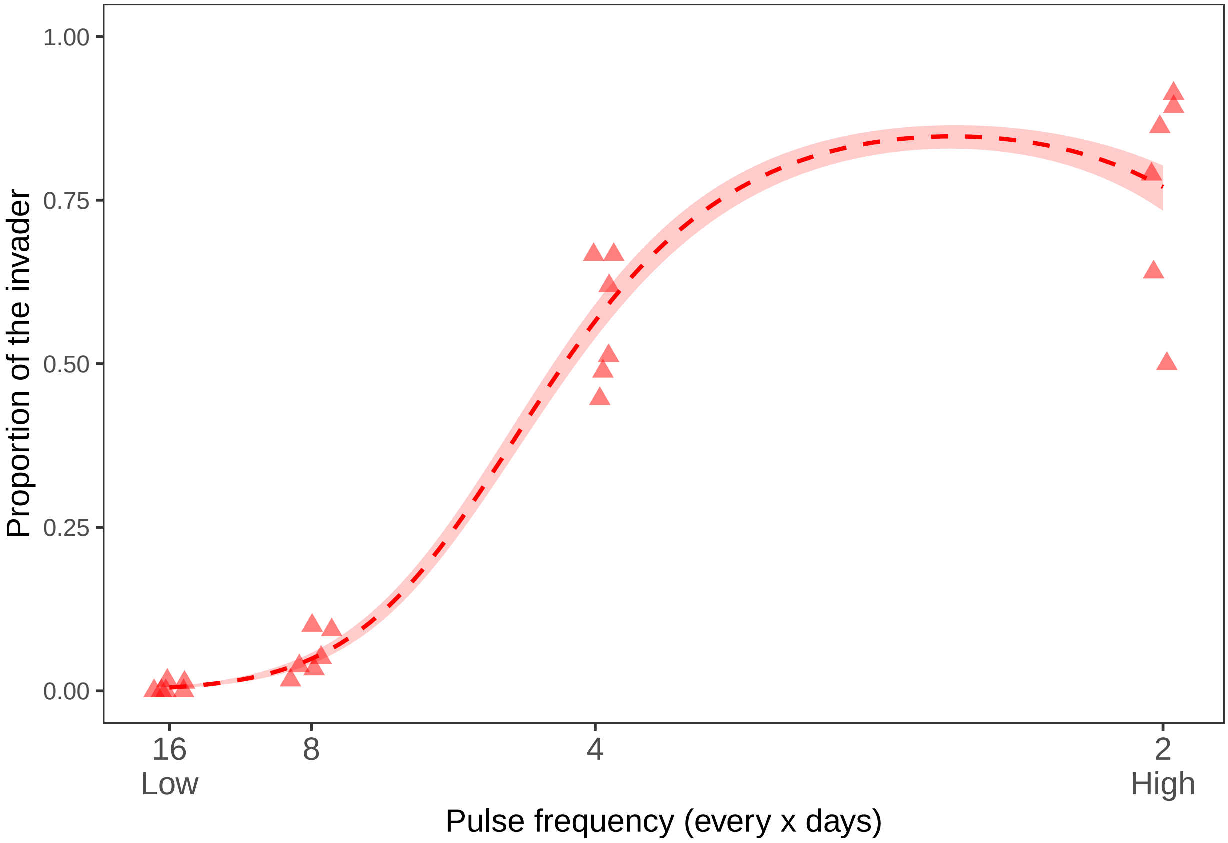
```
# Issues with over- and under-estimating the data... need to include another term as expected.

# Second attempt at the logistic regression: include disturbance^2 as a covariate.
fit.inv.prop2 = glm(
  cbind(inv, (p+o+s+a+v)) ~ disturbance + I(disturbance^2),
  data = data.diversity,
  subset = (invaded=="yes"),
  family = "binomial")

# Model diagnostics
autoplot.custom(fit.inv.prop2)
```



```
# The Q-Q plot could be better.

drop1(fit.inv.prop2, test="LRT")
```

```
## Single term deletions
##
## Model:
## cbind(inv, (p + o + s + a + v)) ~ disturbance + I(disturbance^2)
##                   Df Deviance    AIC    LRT  Pr(>Chi)
## <none>                  43.80 118.46
## disturbance        1   335.50 408.16 291.69 < 2.2e-16 ***
## I(disturbance^2)   1   191.64 264.30 147.84 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# This model is much better than the first one.

# Make a dataframe to calculate the model predictions +/- the standard errors.
pred.inv.prop2 = predict(fit.inv.prop2, newdata=newdat, type="link", se.fit=T) %>%
  magrittr::extract(c("fit", "se.fit")) %>%
  as.data.frame() %>%
  cbind(newdat, .) %>%
  transform(p.fit = plogis(fit), p.upr=plogis(fit+se.fit), p.lwr=plogis(fit-se.fit))

# FIGURE 3A
# Plot the model predictions onto the data points.
fig.inv.prop +
  geom_line(data=pred.inv.prop2, aes(y=p.fit), size=0.75, linetype=2, color="red") +
  geom_ribbon(data=pred.inv.prop2,
    aes(y=NULL, ymin=p.lwr, ymax=p.upr), alpha=0.2, color=NA, fill="red")
```



```
ggsave("inv_prop.png")
```

```
## Saving 6.5 x 4.5 in image
```

```
# The fit looks much better!
# The model output.
summary(fit.inv.prop2)
```

```
##
## Call:
## glm(formula = cbind(inv, (p + o + s + a + v)) ~ disturbance +
##     I(disturbance^2), family = "binomial", data = data.diversity,
##     subset = (invaded == "yes"))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8782  -0.8796  -0.1900   1.3030   1.8057
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -8.0313     0.5155  -15.58   <2e-16 ***
## disturbance       47.8732     3.5457   13.50   <2e-16 ***
## I(disturbance^2) -58.7848     5.3586  -10.97   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 750.237  on 23  degrees of freedom
## Residual deviance:  43.804  on 21  degrees of freedom
## AIC: 118.46
##
## Number of Fisher Scoring iterations: 5
```

```r
# Calculate McFadden's pseudo R-squared:
##  1 - (deviance)/null.deviance
with(fit.inv.prop2, 1 - (deviance)/null.deviance)
```

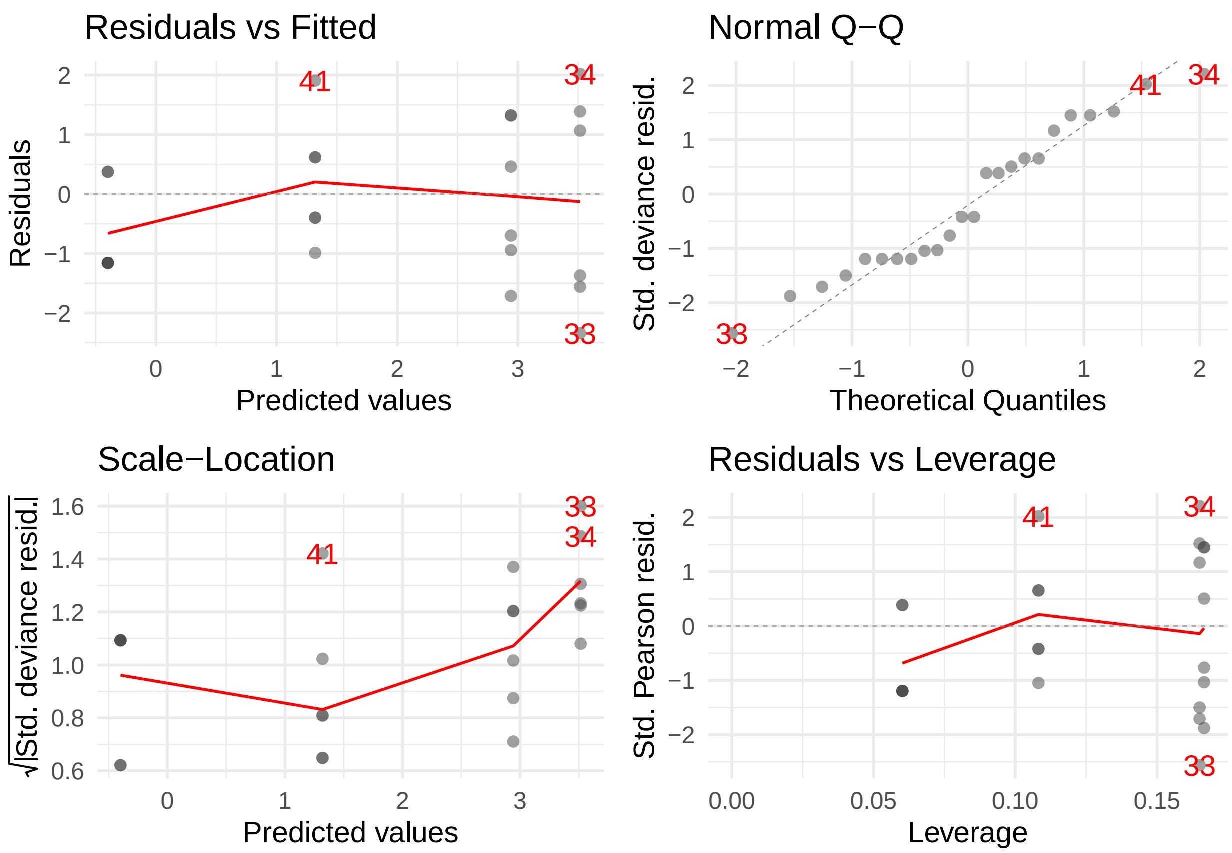
```
## [1] 0.9416137
```

It looks like `pulse frequency` affects the proportion of the invader in the community.

3.2 Density of the invader.

Here we assess the effect of `pulse frequency` on the invader density.

```r
fig.inv.density = ggplot(data=subset(data.diversity, invaded=="yes"),
 mapping=aes(x=disturbance, y=inv/total_dilution)) +
 # geom_boxplot(aes(group=disturbance), outlier.shape = NA, lwd = 0.5,
 #   width = 0.03, show.legend = F, color="red") +
  geom_jitter(position=position_jitter(width=0.01, seed=21, height=0),
    alpha=0.5, size=2.5, color="red", shape=17, fill="red") +
  scale_y_continuous(label=scientific_10) +
  ylab("Invader density (CFU per microcosm)") +
  fig.layer +
  pub.layer +
  theme(legend.position="none")


# Attempt 1: Poisson regression.
fit.inv.density = glm(
  inv ~ disturbance + I(disturbance^2) + offset(log(total_dilution)),
  data=data.diversity,
  subset=(invaded=="yes"),
  family="poisson")

# Model diagnostics
autoplot.custom(fit.inv.density)
```



```
# Not too bad. The Q-Q plot could be better. Scale-Location plot shows some overdispersion?

drop1(fit.inv.density, test="LR")
```

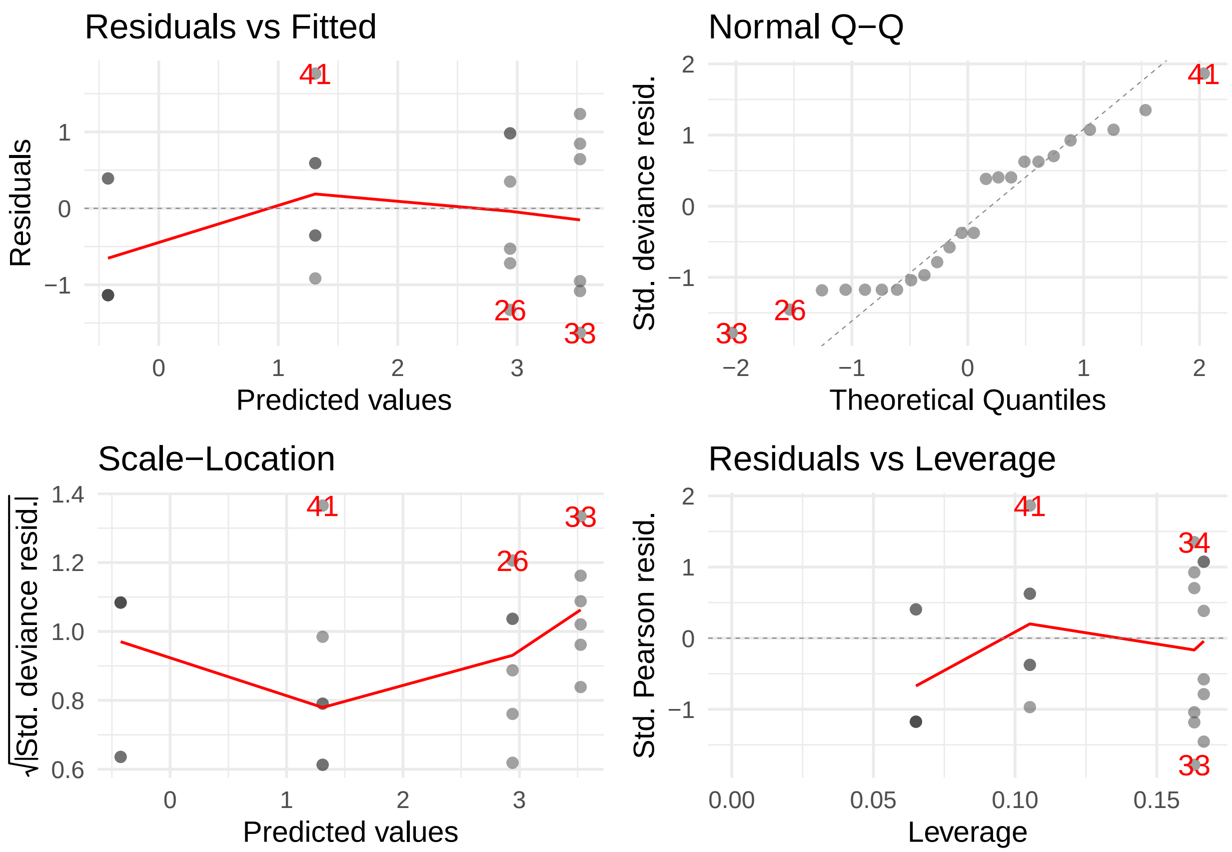
```
## Single term deletions
##
## Model:
## inv ~ disturbance + I(disturbance^2) + offset(log(total_dilution))
##                  Df Deviance    AIC    LRT  Pr(>Chi)
## <none>                 36.31 126.18
## disturbance       1   317.02 404.89 280.71 < 2.2e-16 ***
## I(disturbance^2)  1   274.81 362.67 238.49 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Better not to drop any term.

summary(fit.inv.density)
```

```
##
## Call:
## glm(formula = inv ~ disturbance + I(disturbance^2) + offset(log(total_dilution)),
##     family = "poisson", data = data.diversity, subset = (invaded ==
##         "yes"))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3413  -1.1586  -0.3975   0.7302   2.0192
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)       15.1558     0.4653   32.57   <2e-16 ***
## disturbance       37.4242     2.9862   12.53   <2e-16 ***
```

```
## I(disturbance^2) -52.9559     4.2291  -12.52   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 363.098  on 23  degrees of freedom
## Residual deviance:  36.313  on 21  degrees of freedom
## AIC: 126.18
##
## Number of Fisher Scoring iterations: 5
```

```
# Slightly overdispersed?


# Attempt 2: Negative binomial regression
fit.inv.density2 = MASS::glm.nb(
  inv ~ disturbance + I(disturbance^2) + offset(log(total_dilution)),
  data = data.diversity,
  subset = (invaded=="yes"),
  link = "log")

# Model diagnostics
autoplot.custom(fit.inv.density2)
```



```
AIC(fit.inv.density, fit.inv.density2)
```

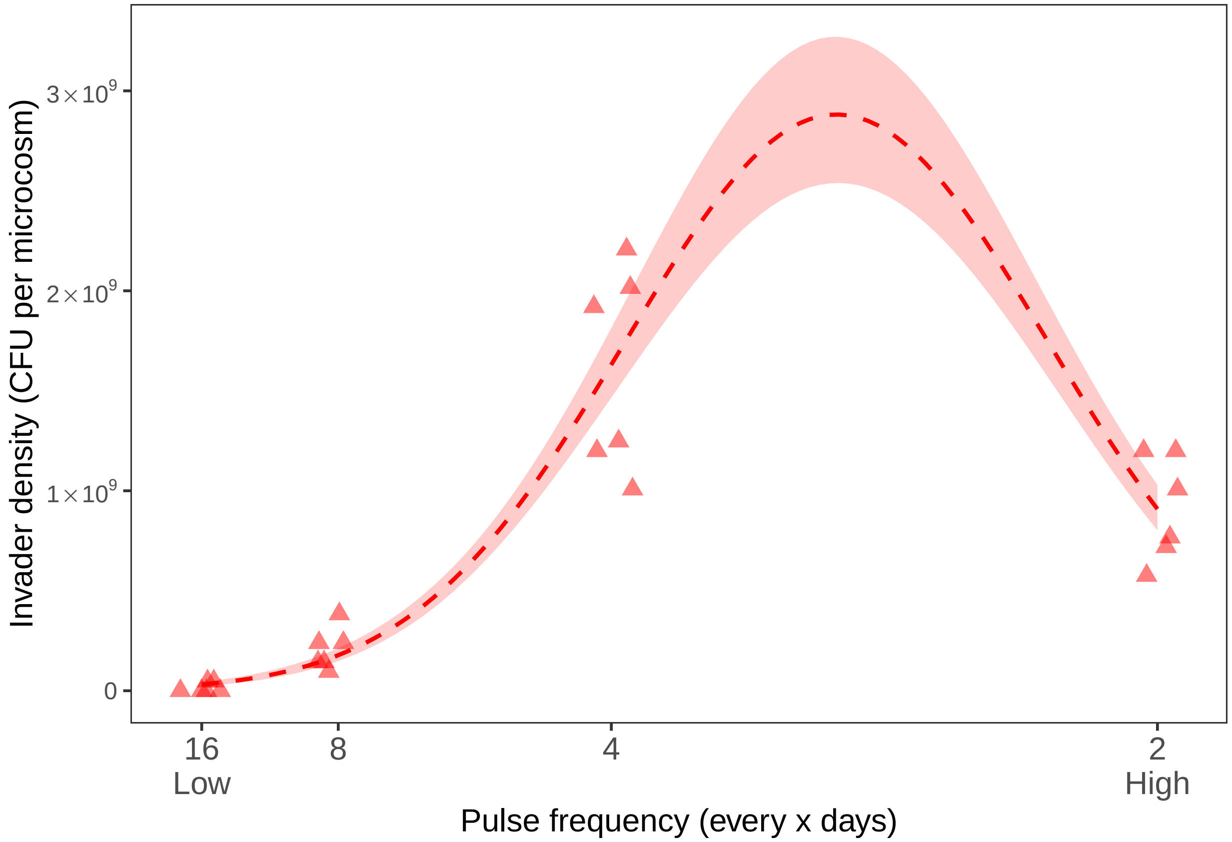
```
##                  df      AIC
## fit.inv.density   3 126.1782
## fit.inv.density2  4 123.6628
```

```
# The negative binomial model is slightly better, but not by much...

# Add another term?
```

```
add1(fit.inv.density2, ~ . + I(disturbance^3), test="LRT")
```

```
## Single term additions
##
## Model:
## inv ~ disturbance + I(disturbance^2) + offset(log(total_dilution))
##                   Df Deviance    AIC    LRT Pr(>Chi)
## <none>                 22.366 121.66
## I(disturbance^3)  1   20.684 121.98 1.6812   0.1948
```

```
# No.

# Plot the model.
pred.inv.density = predict(fit.inv.density2, newdata=newdat, type="link", se.fit=T) %>%
  magrittr::extract(c("fit", "se.fit")) %>%
  as.data.frame() %>%
  cbind(newdat, .) %>%
  transform(upr=fit+se.fit, lwr=fit-se.fit) %>%
  transform(r.fit=exp(fit), r.upr=exp(upr), r.lwr=exp(lwr))

# FIGURE 3B
fig.inv.density -
  geom_line(data=pred.inv.density, aes(y=r.fit),
    size=0.75, linetype=2, color="red") -
  geom_ribbon(data=pred.inv.density, aes(y=NULL, ymin=r.lwr, ymax=r.upr),
    alpha=0.2, color=NA, fill="red")
```



```
ggsave("fig_density.png")
```

```
## Saving 6.5 x 4.5 in image
```

```
# The model output.
summary(fit.inv.density2)
```

```
##
## Call:
## MASS::glm.nb(formula = inv ~ disturbance + I(disturbance^2) +
##     offset(log(total_dilution)), data = data.diversity, subset = (invaded ==
##     "yes"), link = "log", init.theta = 24.97083061)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6287  -1.0958  -0.3555   0.6042   1.7638
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)       15.1086     0.5053   29.90   <2e-16 ***
## disturbance       37.7925     3.4210   11.05   <2e-16 ***
## I(disturbance^2) -53.5095     5.0078  -10.69   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(24.9708) family taken to be 1)
##
##     Null deviance: 247.136  on 23  degrees of freedom
## Residual deviance:  22.366  on 21  degrees of freedom
## AIC: 123.66
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  25.0
##           Std. Err.:  19.0
##
##  2 x log-likelihood:  -115.663
```

```
# Calculate McFadden's pseudo R-squared:
##  1 - (deviance)/null.deviance
with(fit.inv.density2, 1 - (deviance)/null.deviance)
```

```
## [1] 0.9095002
```

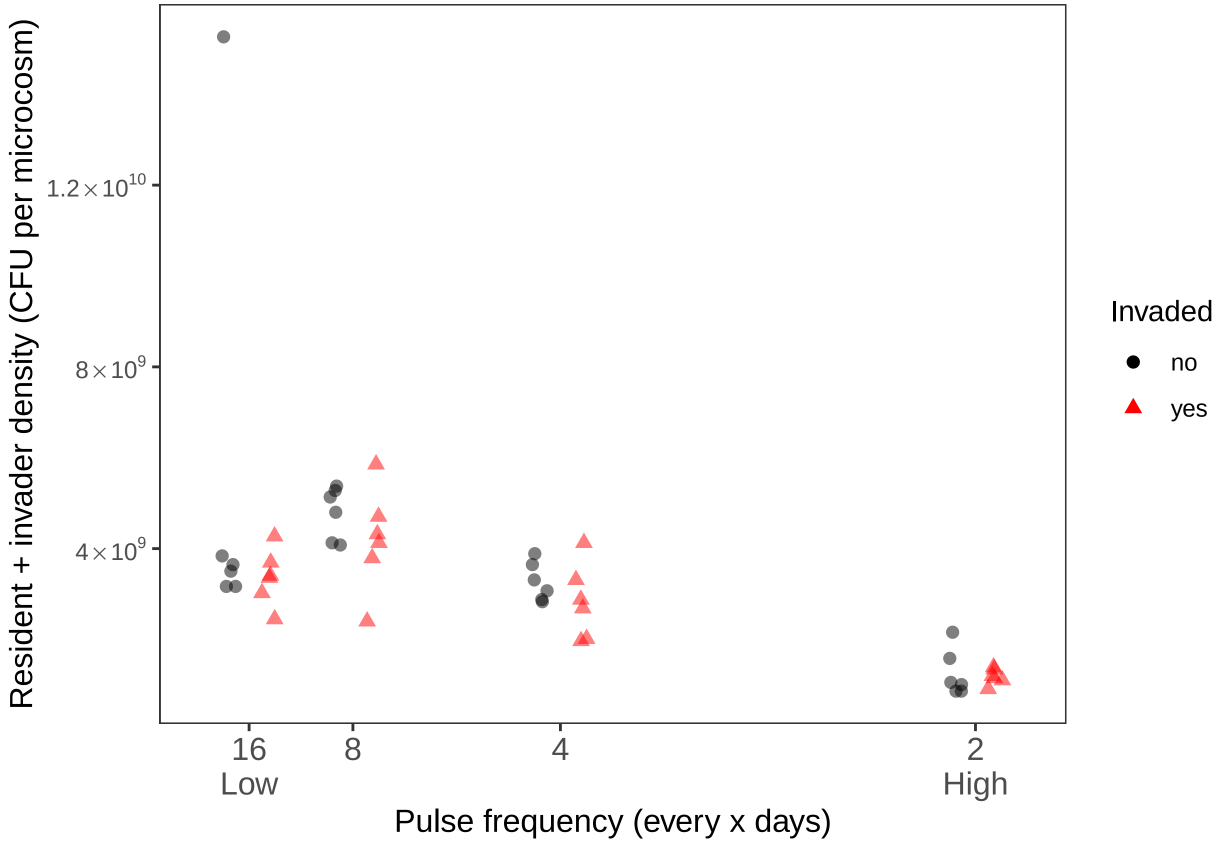
It looks like the `pulse frequency` affects the density of the invader.

4 Density of the community.

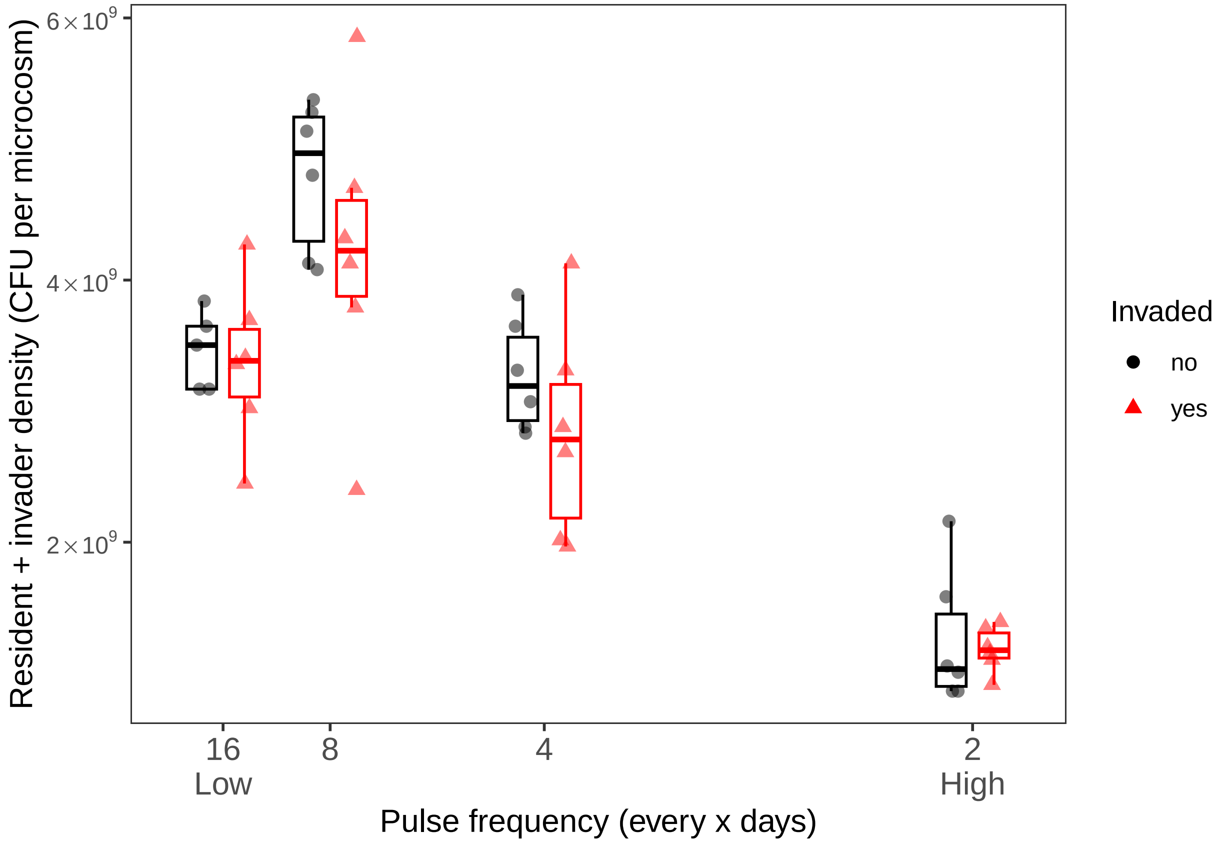
4.1 The effects of `pulse frequency` on the total (residents + invader) density

Here we test whether the `invasion` affects the total (resident + invader) density of the community.

```
fig.total.density = ggplot(data=data.diversity,
    mapping = aes(x=disturbance, y = (p+o+s+a+v+inv) / total_dilution,
    color=invaded, shape=invaded)) +
  geom_point(position = position_jitterdodge(jitter.width = 0.01,
      jitter.height = 0, dodge.width = 0.05, seed=1),
      alpha=0.5, size=2) +
  scale_y_continuous(label=scientific_10) +
  ylab("Resident + invader density (CFU per microcosm)") +
  fig.layer +
  pub.layer


fig.total.density
```



```
# There's a clear outlier, so remove it.
fig.total.density = fig.total.density %+%
  subset(data.diversity, (p + o + s + a + v + inv)/total_dilution <= 1e10)
# It looks like the total density changes with pulse frequency, but not with invasion.

# FIGURE 4B
fig.total.density +
  geom_boxplot(
    aes(group = interaction(disturbance, invaded)),
    outlier.shape = NA, lwd = 0.5,
    position = position_dodge(width = 0.05),
    width = 0.035, show.legend = F)
```



```
ggsave("total_density.png")
```

```
## Saving 6.5 x 4.5 in image
```

```
# Testing the effects of invasion as a two-way ANOVA.
fit.total.density = MASS::glm.nb(
  I(p+o+s+a+v+inv) ~ as.factor(disturbance) * invaded +
    offset(log(total_dilution)),
  data=data.diversity,
  subset=(p + o + s + a + v + inv)/total_dilution <= 1e10,
  link="log")

car::Anova(fit.total.density, test="LR")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: I(p + o + s + a + v + inv)
##                                  LR Chisq Df Pr(>Chisq)
## as.factor(disturbance)            242.299  3    < 2e-16 ***
## invaded                             2.901  1    0.08854 .
## as.factor(disturbance):invaded      0.751  3    0.86116
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Invasion has slightly significant effect on total density (p=0.09)

# The model output.
summary(fit.total.density)
```

```
##
## Call:
## MASS::glm.nb(formula = I(p + o + s + a + v + inv) ~ as.factor(disturbance) *
##     invaded + offset(log(total_dilution)), data = data.diversity,
##     subset = (p + o + s + a + v + inv)/total_dilution <= 1e+10,
```

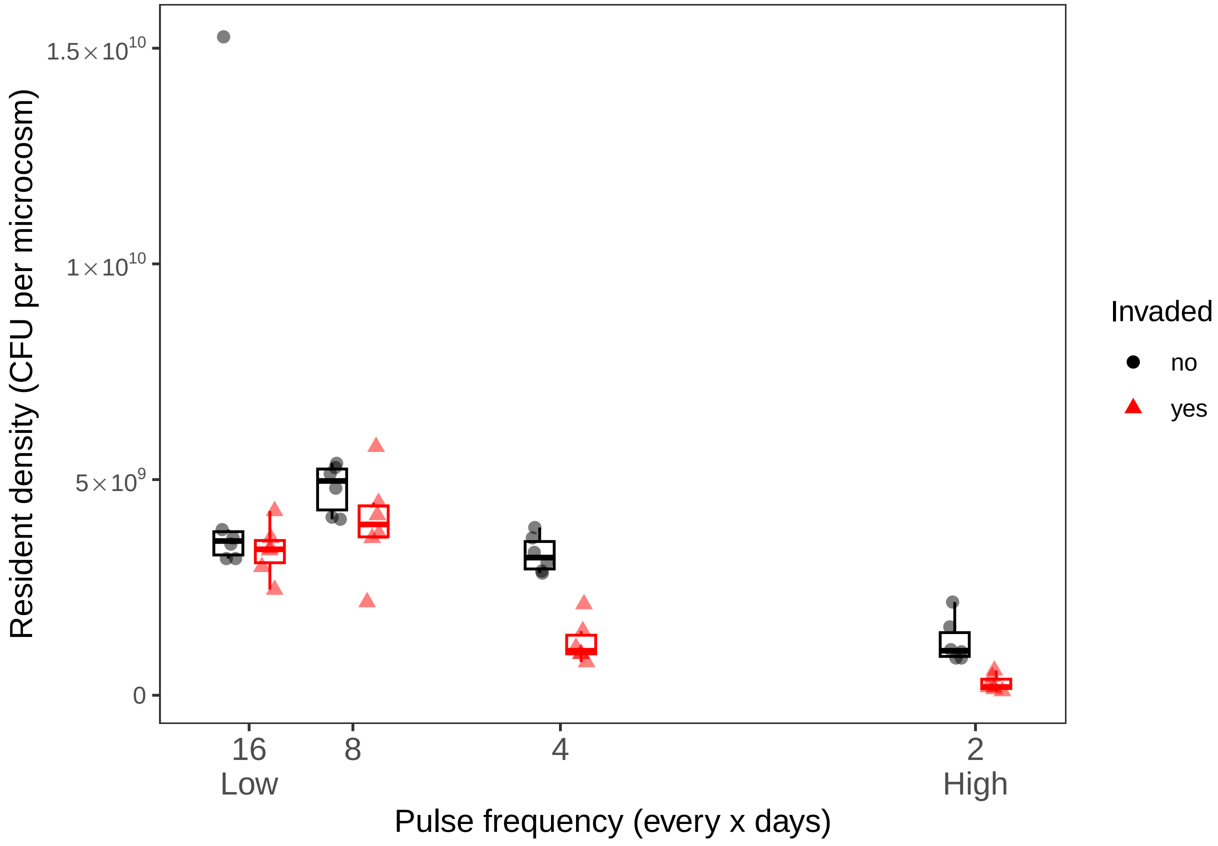
```
##     link = "log", init.theta = 48.2239598)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7503  -0.6696   0.0000   0.5601   2.5922
##
## Coefficients:
##                                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)                             21.96615    0.08317 264.109  < 2e-16 ***
## as.factor(disturbance)0.125              0.32573    0.10973   2.969  0.00299 **
## as.factor(disturbance)0.25              -0.05748    0.11322  -0.508  0.61164
## as.factor(disturbance)0.5               -1.01495    0.12939  -7.844 4.37e-15 ***
## invadedyes                              -0.02857    0.11291  -0.253  0.80027
## as.factor(disturbance)0.125:invadedyes -0.10496    0.15242  -0.689  0.49105
## as.factor(disturbance)0.25:invadedyes  -0.11585    0.15790  -0.734  0.46312
## as.factor(disturbance)0.5:invadedyes   -0.03047    0.18107  -0.168  0.86638
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(48.224) family taken to be 1)
##
##     Null deviance: 294.27  on 46  degrees of freedom
## Residual deviance:  48.28  on 39  degrees of freedom
## AIC: 379.69
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  48.2
##           Std. Err.:  18.5
##
##  2 x log-likelihood:  -361.686
```

Invasion doesn't seem to affect the total community density. Does an increase in invader density lead to a decrease in the resident density?

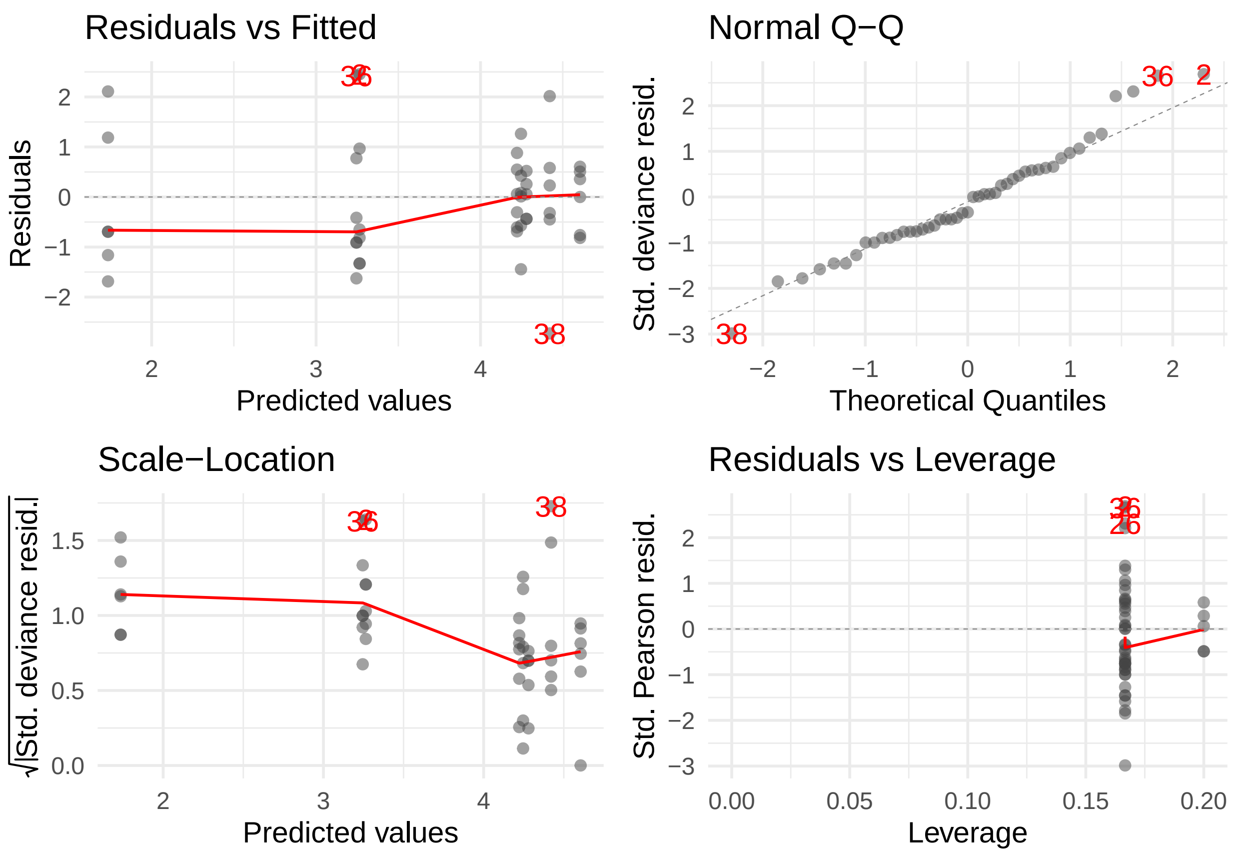
4.2 The effects of `invasion` on the resident community density.

The analyses above suggest that the total densities do not differ between invaded and uninvaded communities. We will test this hypothesis below.

```
fig.res.density = ggplot(data=data.diversity,
  mapping=aes(x=disturbance, y=(p+o+s+a+v)/total_dilution,
    color=invaded, shape=invaded)) +
  geom_boxplot(
    aes(group = interaction(disturbance, invaded)),
    outlier.shape = NA, lwd = 0.5,
    position = position_dodge(width = 0.05),
    width = 0.035, show.legend = F) +
  geom_point(position = position_jitterdodge(jitter.width = 0.01,
    jitter.height = 0, dodge.width = 0.05, seed=1),
    alpha=0.5, size=2) +
  ylab("Resident density (CFU per microcosm)") +
  scale_y_continuous(label=scientific_10) +
  fig.layer +
  pub.layer

fig.res.density
```



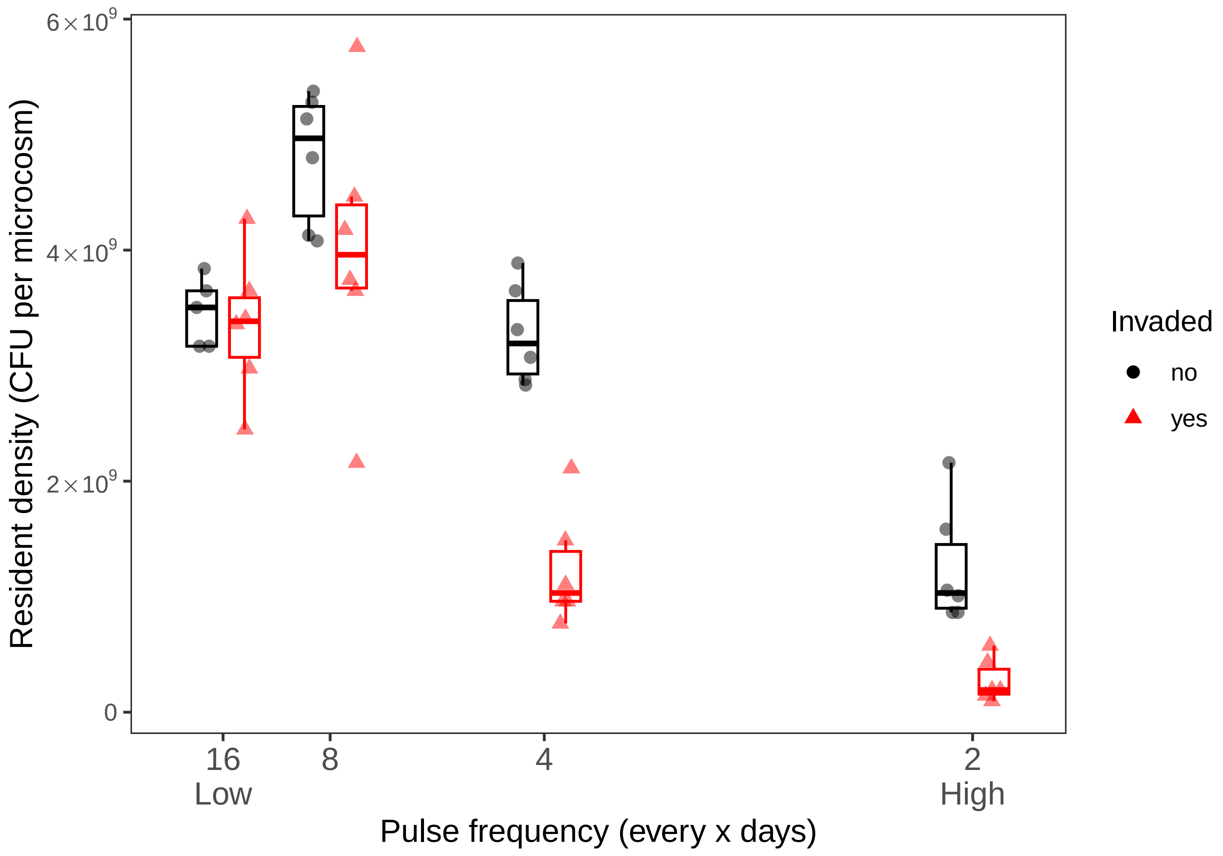
```
# One of the data points is clearly an outlier. Subset the data for the analyses.

fig.res.density = fig.res.density %+%
  subset(data.diversity, (p + o + s + a + v + inv)/total_dilution <= 1e10)
# The model that best captures the effects of disturbance on the community density is complex.
# So we will perform a simple two-way ANOVA to see if the invaded communities differ
# from uninvaded ones.

fit.res.density = MASS::glm.nb(
  I(p+o+s+a+v) ~ invaded*as.factor(disturbance) +
    offset(log(total_dilution)),
  data=data.diversity,
  subset=(p + o + s + a + v + inv)/total_dilution <= 1e10,
  link="log")

# Model diagnostics
autoplot.custom(fit.res.density)
```



```
# The Q-Q plot doesn't look too bad.

# Two-way ANOVA.
car::Anova(fit.res.density, test="LR")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: I(p + o + s + a + v)
##                                 LR Chisq Df Pr(>Chisq)
## invaded                            57.50  1  3.377e-14 ***
## as.factor(disturbance)            364.10  3  < 2.2e-16 ***
## invaded:as.factor(disturbance)     62.05  3  2.140e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Significant effects of invasion and the interaction between invasion and disturbance.

# FIGURE 4A
fig.res.density
```



```
ggsave("res_density.png")
```

```
## Saving 6.5 x 4.5 in image
```

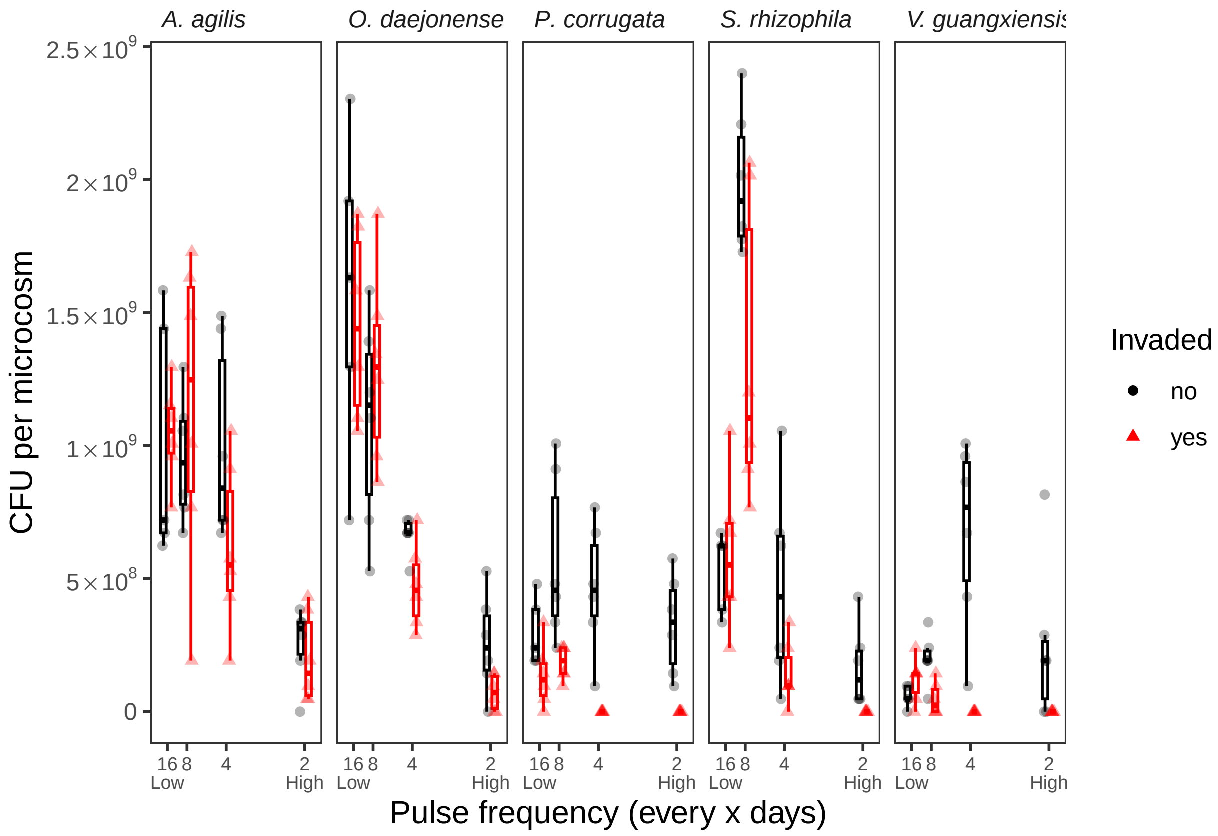
`Invasion` and `pulse frequency`×`invasion` affect the resident density.

5 Multivariate analyses on the individual resident species using Latent Variable Models.

Here we use the `gllvm` package to perform multivariate analyses on the density of each resident species.

```
library(gllvm)

# Labels for the species' names to be used in the plots.
labs = as_labeller(c(
    `p` = "P. corrugata",
    `o` = "O. daejonense",
    `s` = "S. rhizophila",
    `a` = "A. agilis",
    `v` = "V. guangxiensis"
))

# Create the data figure.
fig.indiv.res = data.diversity %>%
  subset((p + o + s + a + v + inv)/total_dilution <= 1e10) %>%
  pivot_longer(cols=p:v, names_to="spp", values_to="cfu") %>%
  ggplot(data=.,
    aes(x = disturbance, y = cfu/total_dilution,
      color = invaded, shape = invaded)) +
    geom_point(position = position_jitterdodge(jitter.width = 0.01,
      jitter.height = 0, dodge.width = 0.05, seed=1),
```

```
      alpha=0.3) +
    facet_wrap(facets = "spp", labeller = labs, nrow=1) +
    ylab("CFU per microcosm") +
    scale_y_continuous(label=scientific_10) +
    fig.layer +
    pub.layer +
    theme(strip.text = element_text(face = "italic", size=9),
      axis.text.x = element_text(size = 6.5))

data.temp = subset(data.diversity, (p + o + s + a + v + inv)/total_dilution <= 1e10)

# Analysis using factor(disturbance)

# FIGURE 5
# Boxplot
fig.indiv.res -
  geom_boxplot(
    aes(group = interaction(disturbance, invaded)),
    outlier.shape = NA, lwd = 0.5,
    position = position_dodge(width = 0.05),
    width = 0.035, show.legend = F)
```



```
    # Changing facet_wrap, otherwise hard to see.
    ggsave("indiv_density.png", width=12)
```

```
## Saving 12 x 4.5 in image
```

```
# Each resident species respond differently to invasion and pulse disturbances.

aicc.gllvm = rep(NA, 3)
gllvm.list = vector(mode="list", length=3)

for(i in 1:length(aicc.gllvm) ){
```

```r
  fit = gllvm(
    y = dplyr::select(data.temp, p:v),
    X = data.temp,
    formula = ~ invaded * as.factor(disturbance),
    offset = data.temp$total_dilution,
    num.lv = i-1,
    family = "negative.binomial", row.eff = "random",
    seed=1)

  gllvm.list[[i]] = fit
  aicc.gllvm[i] = summary(fit)$AICc
}

# Analysis with the model with the lowest AICc.
# Find the model with the lowest AICc.
which.min(aicc.gllvm)-1
```

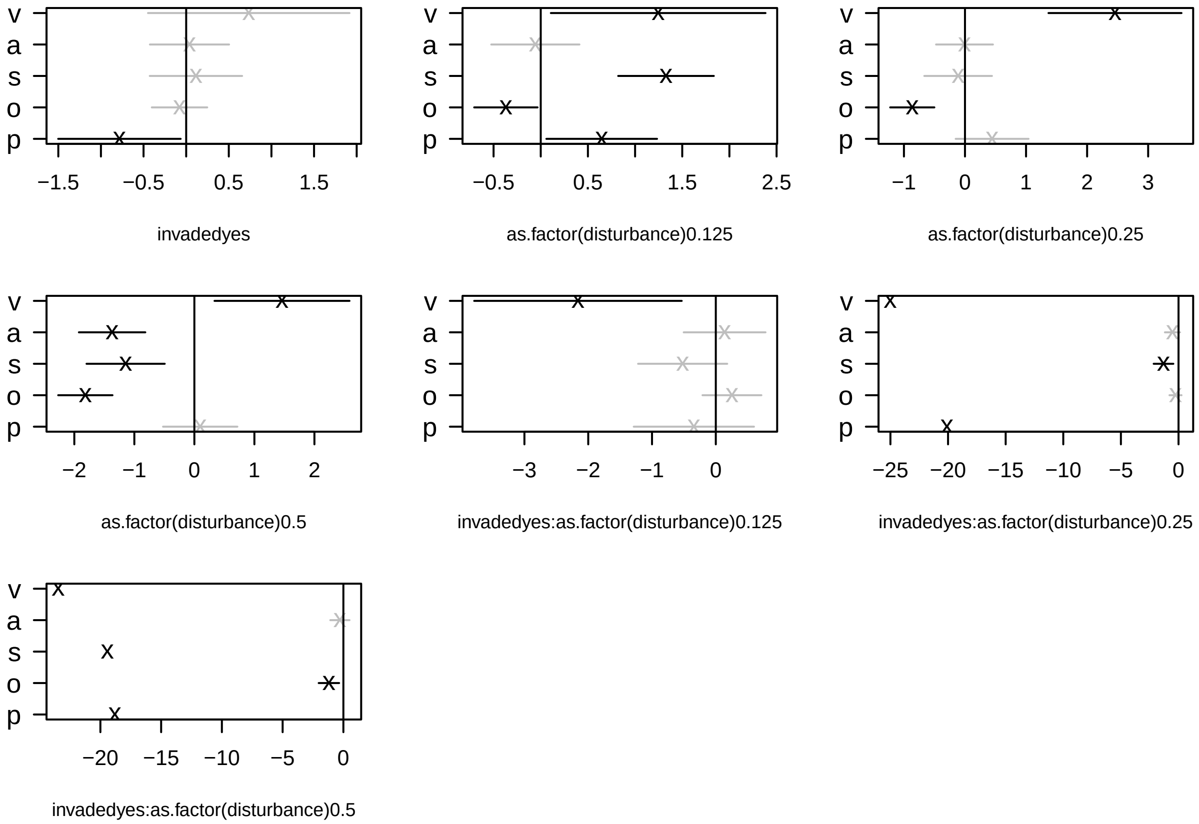
```
## [1] 0
```

```r
# Number of latent variables: 0

# Set the model.
fit.gllvm0 = gllvm.list[[which.min(aicc.gllvm)]]

coefplot(fit.gllvm0, cex.ylab=1.2, cex.xlab=0.9,
  cex=1.25, order=F, mar=c(4, 3, 2, 1),
  mfrow=c(3,3))

summary(fit.gllvm0)
```

```
##
## Call:
## gllvm(y = dplyr::select(data.temp, p:v), X = data.temp, formula = ~invaded *
##     as.factor(disturbance), family = "negative.binomial", num.lv = i -
##     1, row.eff = "random", offset = data.temp$total_dilution,
##     seed = 1)
##
## Family:  negative.binomial
##
## AIC:  1259.366 AICc:  1282.366 BIC:  1418.507 LL:  -583.7 df:  46
##
## Informed LVs:  0
## Constrained LVs:  0
## Unconstrained LVs:  0
##
## Formula:  ~invaded * as.factor(disturbance)
## LV formula:  ~yNULL
##
## Coefficients predictors:
##                                            Estimate Std. Error    z value
## invadedyes:p                             -7.831e-01  3.665e-01 -2.137e+00
## invadedyes:o                             -7.817e-02  1.659e-01 -4.710e-01
## invadedyes:s                              1.144e-01  2.764e-01  4.140e-01
## invadedyes:a                              3.891e-02  2.369e-01  1.640e-01
## invadedyes:v                              7.340e-01  6.030e-01  1.217e+00
## as.factor(disturbance)0.125:p             6.464e-01  2.993e-01  2.160e+00
## as.factor(disturbance)0.125:o            -3.695e-01  1.714e-01 -2.156e+00
## as.factor(disturbance)0.125:s             1.328e+00  2.587e-01  5.133e+00
## as.factor(disturbance)0.125:a            -5.716e-02  2.385e-01 -2.400e-01
## as.factor(disturbance)0.125:v             1.245e+00  5.805e-01  2.144e+00
```

```
## as.factor(disturbance)0.25:p              4.441e-01  3.045e-01  1.459e+00
## as.factor(disturbance)0.25:o             -8.633e-01  1.846e-01 -4.678e+00
## as.factor(disturbance)0.25:s             -1.121e-01  2.826e-01 -3.970e-01
## as.factor(disturbance)0.25:a             -7.972e-03  2.377e-01 -3.400e-02
## as.factor(disturbance)0.25:v              2.457e+00  5.557e-01  4.421e+00
## as.factor(disturbance)0.5:p               9.726e-02  3.160e-01  3.080e-01
## as.factor(disturbance)0.5:o              -1.816e+00  2.308e-01 -7.871e+00
## as.factor(disturbance)0.5:s              -1.145e+00  3.324e-01 -3.445e+00
## as.factor(disturbance)0.5:a              -1.371e+00  2.824e-01 -4.854e+00
## as.factor(disturbance)0.5:v               1.460e+00  5.738e-01  2.544e+00
## invadedyes:as.factor(disturbance)0.125:p -3.441e-01  4.808e-01 -7.160e-01
## invadedyes:as.factor(disturbance)0.125:o  2.531e-01  2.356e-01  1.074e+00
## invadedyes:as.factor(disturbance)0.125:s -5.199e-01  3.565e-01 -1.458e+00
## invadedyes:as.factor(disturbance)0.125:a  1.378e-01  3.271e-01  4.210e-01
## invadedyes:as.factor(disturbance)0.125:v -2.161e+00  8.299e-01 -2.604e+00
## invadedyes:as.factor(disturbance)0.25:p  -2.010e+01  7.704e-09 -2.609e+09
## invadedyes:as.factor(disturbance)0.25:o  -2.631e-01  2.664e-01 -9.880e-01
## invadedyes:as.factor(disturbance)0.25:s  -1.302e+00  4.354e-01 -2.990e+00
## invadedyes:as.factor(disturbance)0.25:a  -5.234e-01  3.354e-01 -1.560e+00
## invadedyes:as.factor(disturbance)0.25:v  -2.502e+01  2.223e-09 -1.126e+10
## invadedyes:as.factor(disturbance)0.5:p   -1.881e+01  1.984e-08 -9.481e+08
## invadedyes:as.factor(disturbance)0.5:o   -1.190e+00  4.293e-01 -2.772e+00
## invadedyes:as.factor(disturbance)0.5:s   -1.943e+01  9.914e-09 -1.960e+09
## invadedyes:as.factor(disturbance)0.5:a   -2.858e-01  4.035e-01 -7.080e-01
## invadedyes:as.factor(disturbance)0.5:v   -2.348e+01  3.888e-09 -6.039e+09
##                                          Pr(>|z|)
## invadedyes:p                             0.032614 *
## invadedyes:o                             0.637395
## invadedyes:s                             0.678940
## invadedyes:a                             0.869529
## invadedyes:v                             0.223533
## as.factor(disturbance)0.125:p            0.030793 *
## as.factor(disturbance)0.125:o            0.031051 *
## as.factor(disturbance)0.125:s            2.85e-07 ***
## as.factor(disturbance)0.125:a            0.810592
## as.factor(disturbance)0.125:v            0.031996 *
## as.factor(disturbance)0.25:p             0.144697
## as.factor(disturbance)0.25:o             2.90e-06 ***
## as.factor(disturbance)0.25:s             0.691510
## as.factor(disturbance)0.25:a             0.973242
## as.factor(disturbance)0.25:v             9.84e-06 ***
## as.factor(disturbance)0.5:p              0.758262
## as.factor(disturbance)0.5:o              3.53e-15 ***
## as.factor(disturbance)0.5:s              0.000572 ***
## as.factor(disturbance)0.5:a              1.21e-06 ***
## as.factor(disturbance)0.5:v              0.010945 *
## invadedyes:as.factor(disturbance)0.125:p 0.474199
## invadedyes:as.factor(disturbance)0.125:o 0.282641
## invadedyes:as.factor(disturbance)0.125:s 0.144756
## invadedyes:as.factor(disturbance)0.125:a 0.673548
## invadedyes:as.factor(disturbance)0.125:v 0.009213 **
## invadedyes:as.factor(disturbance)0.25:p   < 2e-16 ***
## invadedyes:as.factor(disturbance)0.25:o  0.323293
## invadedyes:as.factor(disturbance)0.25:s  0.002794 **
## invadedyes:as.factor(disturbance)0.25:a  0.118671
## invadedyes:as.factor(disturbance)0.25:v   < 2e-16 ***
## invadedyes:as.factor(disturbance)0.5:p    < 2e-16 ***
## invadedyes:as.factor(disturbance)0.5:o   0.005563 **
## invadedyes:as.factor(disturbance)0.5:s    < 2e-16 ***
## invadedyes:as.factor(disturbance)0.5:a   0.478769
```

```
## invadedyes:as.factor(disturbance)0.5:v    < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

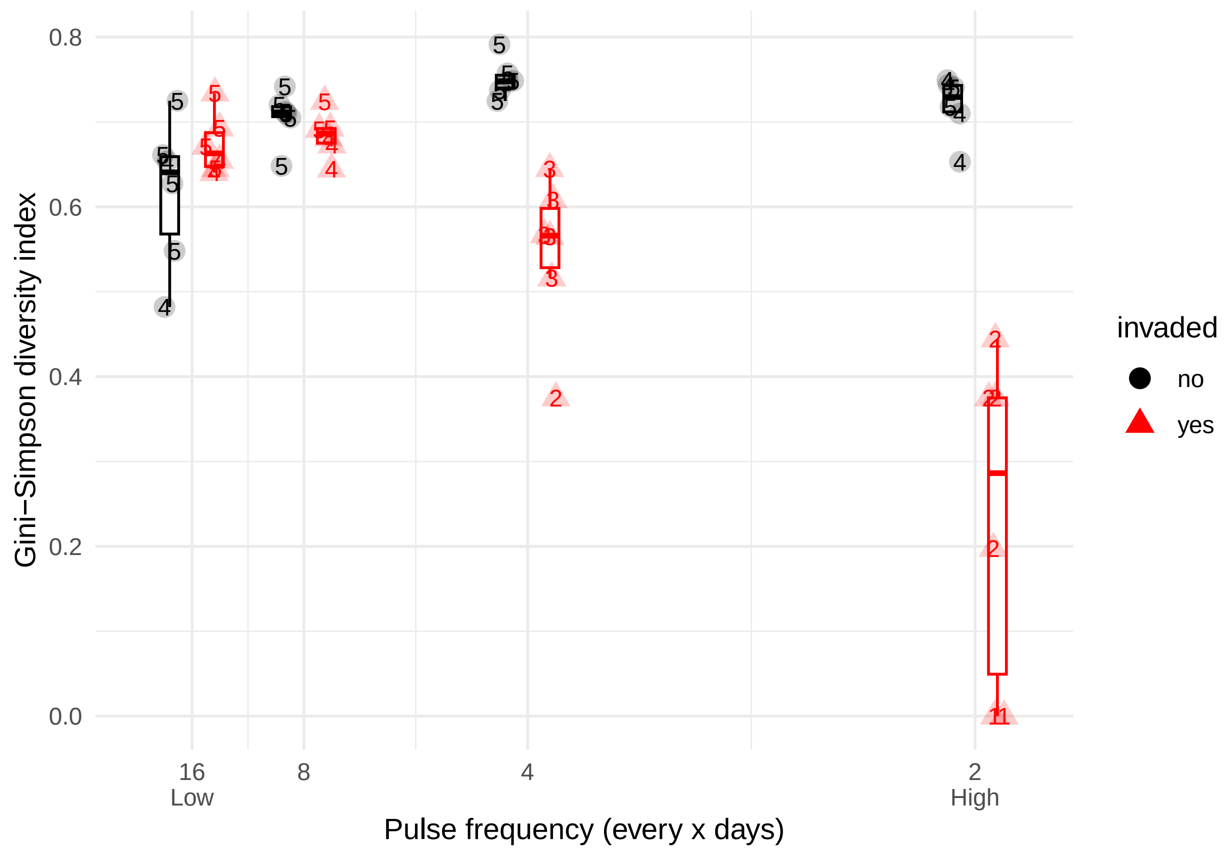


Appendix: Additional analyses

A The effects of `invasion` and `pulse frequency` on the Gini-Simpson index of the resident community diversity.

The following figure shows the effects of `pulse frequency` on the diversity of the resident community, without (black circle) and with (red triangle) an `invasion`. Light background points are individual sample measurements, and the dark points and their error bars are their mean and the standard error of the mean. The number in each sample measurements is the species richness (i.e. the number of detected species; lower detection limit in this experiment: 8×10^6cfu/ml).

```
fig.simpson = ggplot(data=data.diversity,
    mapping=aes(x=disturbance, y=simpson, color=invaded, shape=invaded)) +
  scale_color_manual(values=c("no"="black", "yes"="red")) +
  scale_fill_manual(values=c("no"="black", "yes"="red")) +
  scale_x_continuous(breaks=1/c(2,4,8,16), labels=c("2\nHigh", "4", "8", "16\nLow")) +
  geom_jitter(alpha=0.2, size=3.5,
    position=position_jitterdodge(dodge.width=0.05, jitter.width=0.01, seed=1)) +
  geom_text(aes(label=alpha), size=3, show.legend=F,
    position=position_jitterdodge(dodge.width=0.05, jitter.width=0.01, seed=1)) +
  xlab("Pulse frequency (every x days)") +
  ylab("Gini-Simpson diversity index") +
  guides(color=guide_legend(override.aes = list(alpha=1)))

fig.simpson -
  geom_boxplot(aes(group=interaction(disturbance, invaded)), outlier.shape=NA,
    position=position_dodge(width=0.05), width=0.02, show.legend=F)
```



It looks like the effect of disturbance on diversity is hump-shaped, and the effect of disturbance on diversity differs between invaded and uninvaded samples.

```
lm.simpson1 = lm(simpson ~ (disturbance + I(disturbance^2)) * invaded, data=data.diversity)
summary(lm.simpson1);
```

```
##
## Call:
## lm(formula = simpson ~ (disturbance + I(disturbance^2)) * invaded,
##     data = data.diversity)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.230035 -0.032878  0.006212  0.033055  0.214409
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  0.55094    0.05683   9.695 2.81e-12 ***
## disturbance                  1.34325    0.53928   2.491   0.0168 *
## I(disturbance^2)            -2.01912    0.91391  -2.209   0.0327 *
## invadedyes                   0.15816    0.08037   1.968   0.0557 .
## disturbance:invadedyes      -1.57920    0.76266  -2.071   0.0446 *
## I(disturbance^2):invadedyes  0.57475    1.29246   0.445   0.6588
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08614 on 42 degrees of freedom
## Multiple R-squared:  0.7901, Adjusted R-squared:  0.7651
## F-statistic: 31.62 on 5 and 42 DF,  p-value: 3.258e-13
```

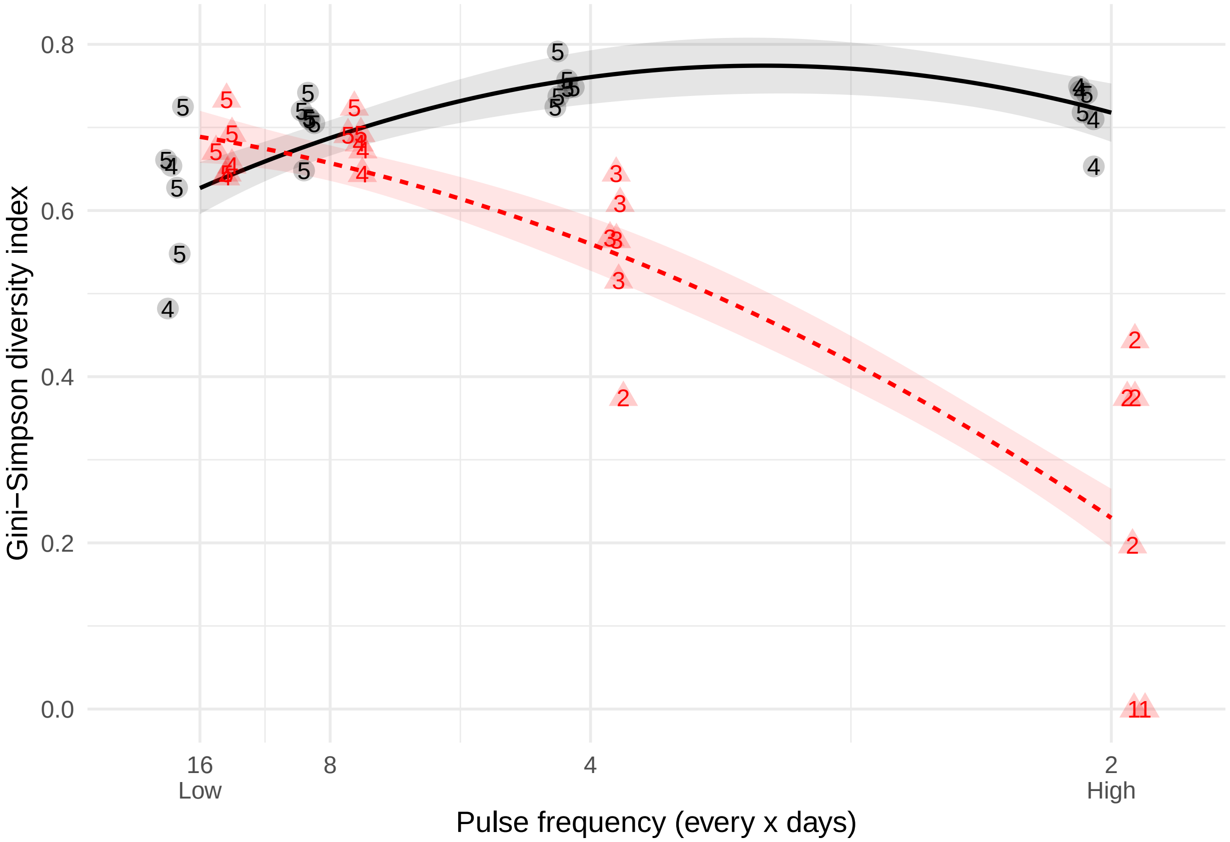
```
# AIC comparison
drop1(lm.simpson1, test="F")
```

```
## Single term deletions
##
## Model:
## simpson ~ (disturbance + I(disturbance^2)) * invaded
##                          Df Sum of Sq     RSS     AIC F value  Pr(>F)
## <none>                                0.31167 -229.78
## disturbance:invaded       1  0.031817 0.34348 -227.11  4.2876 0.04457 *
## I(disturbance^2):invaded  1  0.001467 0.31314 -231.55  0.1978 0.65882
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

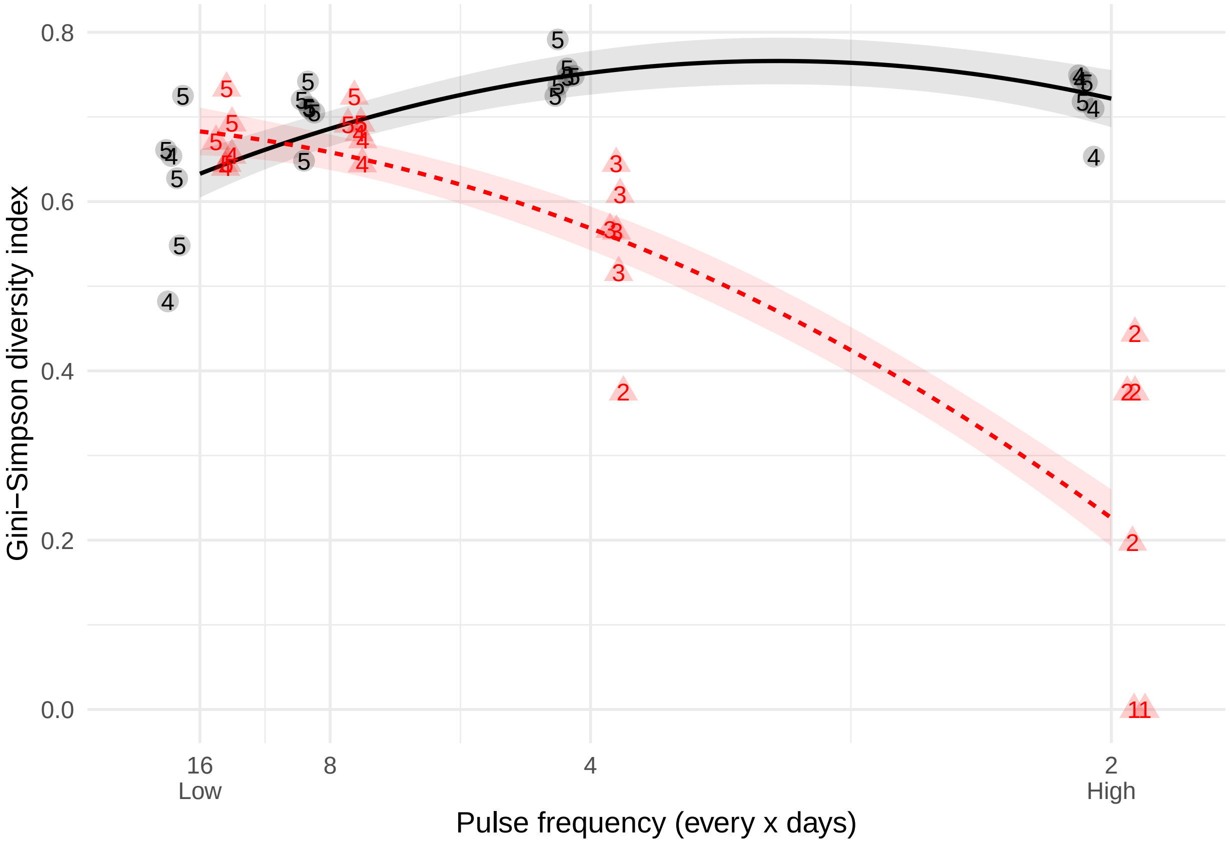
```
# Model with a lower AIC, but not significantly different from the full model.
lm.simpson2 = update(lm.simpson1, ~ . - I(disturbance^2):invaded)
summary(lm.simpson2)
```

```
##
## Call:
## lm(formula = simpson ~ disturbance + I(disturbance^2) + invaded +
##     disturbance:invaded, data = data.diversity)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.226228 -0.027544  0.007554  0.033344  0.218216
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)             0.56607    0.04509  12.554 5.64e-16 ***
## disturbance             1.17692    0.38484   3.058  0.00382 **
## I(disturbance^2)       -1.73174    0.64017  -2.705  0.00975 **
## invadedyes              0.12790    0.04236   3.020  0.00425 **
## disturbance:invadedyes -1.24654    0.14702  -8.479 1.01e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08534 on 43 degrees of freedom
## Multiple R-squared:  0.7891, Adjusted R-squared:  0.7695
## F-statistic: 40.23 on 4 and 43 DF,  p-value: 5.254e-14
```

```
# Plot using the full model.
# Make a dataframe to calculate the model predictions +/- the standard errors.
pred = predict(lm.simpson1, newdata=newdat, type="response", se.fit=T) %>%
  magrittr::extract(c("fit", "se.fit")) %>%
  as.data.frame() %>%
  cbind(newdat, .) %>%
  transform(lwr=fit-se.fit, upr=fit+se.fit)

# FIGURE S1A
fig.simpson -
  geom_line(data=pred, aes(y=fit, linetype=invaded), size=0.75) -
  geom_ribbon(data=pred, aes(y=NULL, ymin=lwr, ymax=upr, fill=invaded),
    show.legend=F, color=NA, alpha=0.1)+
  theme(legend.position="none")
```



```
# Plot using the model without I(disturbance^2):invaded.
# Make a dataframe to calculate the model predictions +/- the standard errors.
pred = predict(lm.simpson2, newdata=newdat, type="response", se.fit=T) %>%
  magrittr::extract(c("fit", "se.fit")) %>%
  as.data.frame() %>%
  cbind(newdat, .) %>%
  transform(lwr=fit-se.fit, upr=fit+se.fit)

# FIGURE S1B
fig.simpson -
  geom_line(data=pred, aes(y=fit, linetype=invaded), size=0.75) -
  geom_ribbon(data=pred, aes(y=NULL, ymin=lwr, ymax=upr, fill=invaded),
    show.legend=F, color=NA, alpha=0.1)+
  theme(legend.position="none")
```


Gini−Simpson diversity index
0.0
0.2
0.4
0.6
0.8
16
Low
8
4
2
High
Pulse frequency (every x days)